

# Software-Defined Real-Time Mesh Networking: Protocol and Experimentation Method

Florian Greff<sup>1,2</sup>, Ye-Qiong Song<sup>2</sup>, Arnaud Samama<sup>1</sup>, Laurent Ciarletta<sup>2</sup>  
<sup>1</sup>Thales Research & Technology, France    <sup>2</sup>LORIA, University of Lorraine, France

## Dynamic Mesh Networking: Bringing Flexibility to Safe Systems

Nowadays, in embedded real-time systems, there is an evolution of needs in terms of computing power and communication between applications. This is due to the changing qualities of sensors, whose data processing needs are increasing (for example in radar applications), and the emergence of new application categories such as multi-sensors. Interaction between applications and sensors is increasing, while features tend to be spread over several computing units. Communication architectures are becoming more complex.

Mesh networking of the components of such systems would reconcile their real-time constraints with the new application needs [1]. The plurality of communication paths should result in increased flexibility, resilience, scalability and load balancing characteristics. From the networking point of view, this implies the ability to dynamically allocate network resources with respect to the needs of running applications. We define this kind of platform as a **Software-Defined Real-time Network (SDRN)**.

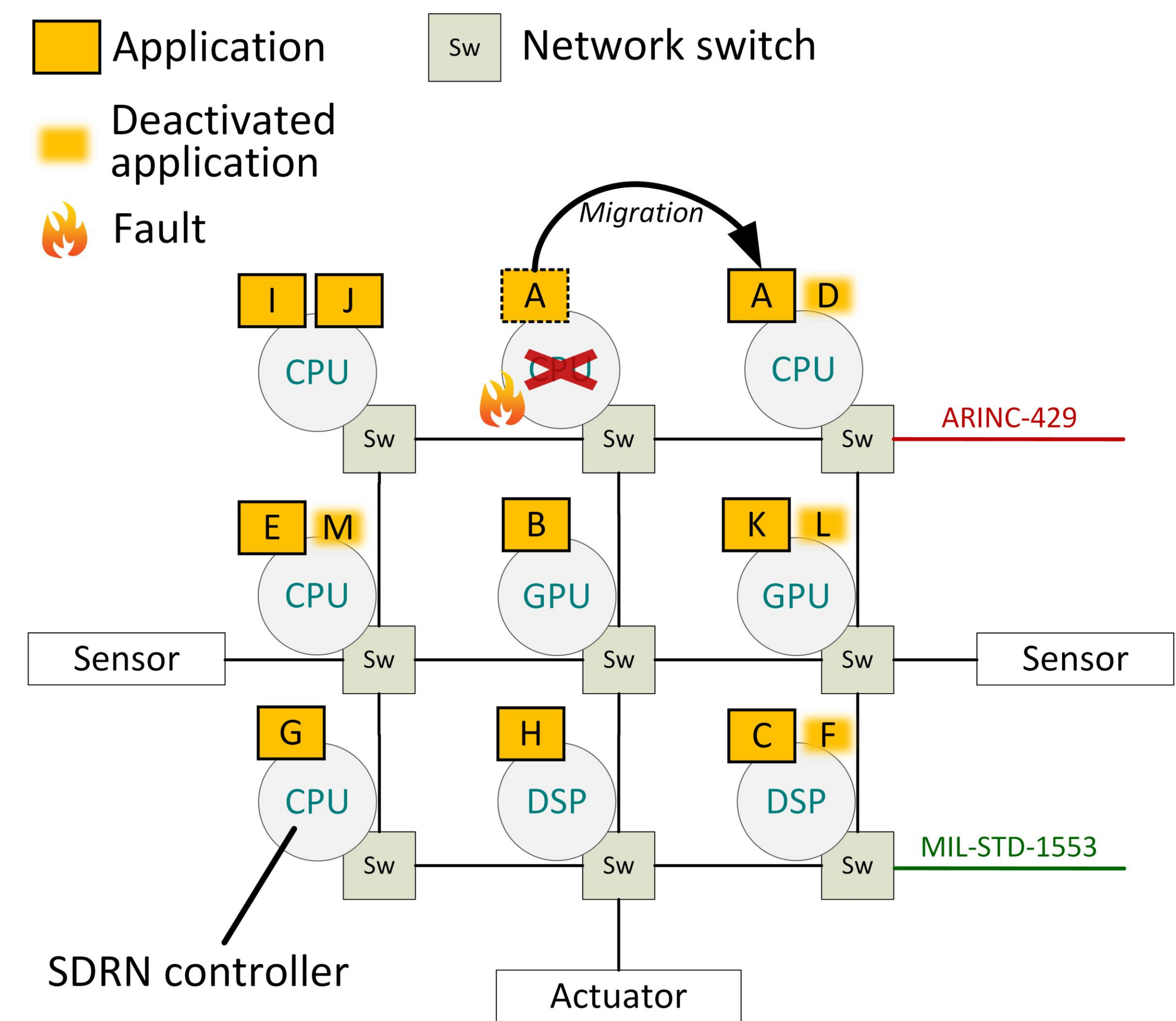


Figure 1. Dynamic system on a mesh network

## First Contribution: a Protocol for the Dynamic Allocation of Network Resources [3]

### Link virtualization

- Each link is divided into sub-channels
- Sub-channels are allotted to flows
- Credit-Based Weighted Round Robin scheduling

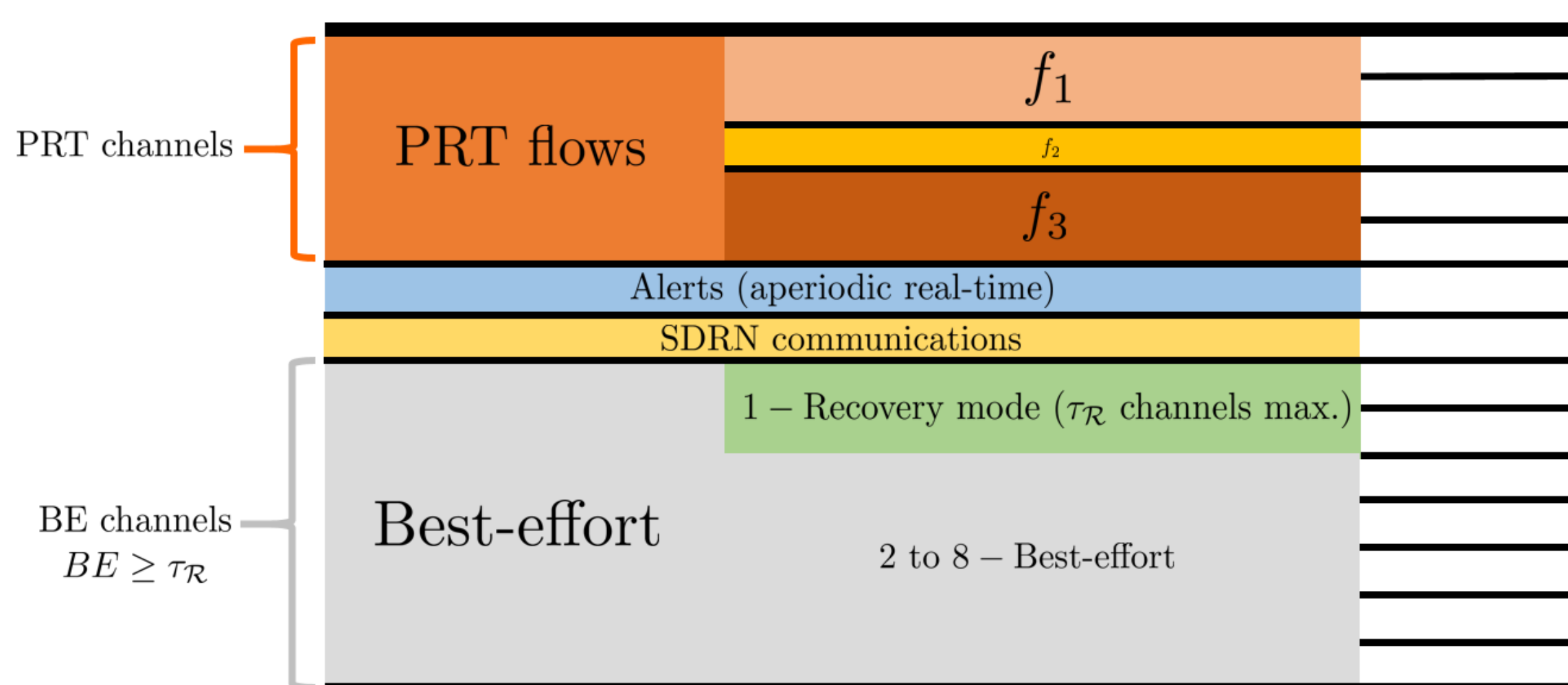


Figure 2. Link virtualization

### Admission control and pathfinding

- From high-level requests:  $req_f = \{T_f, S_f, dl_f, src_f, dest_f\}$
- Take an optimization criterion into account (e.g. load balancing)

### Fault tolerance: hybrid routing

- In **nominal mode**, routing is source-based
- In **recovery mode**, routing is hop-by-hop
- Sub-channels are reserved for flows in recovery mode

### Ongoing work

- In-silicon implementation
- Using the SDN paradigm for node configuration

## Second Contribution: ERICA, a Framework for Conducting Hybrid Experiments

ERICA helps building hybrid experimentation stacks in order to assess new network mechanisms [2]. Adaptation mechanisms allow to build a custom experimentation stack and to run the same application layer on top of different kind of networks (either real or simulated), thus maximizing relevance, portability and reusability.

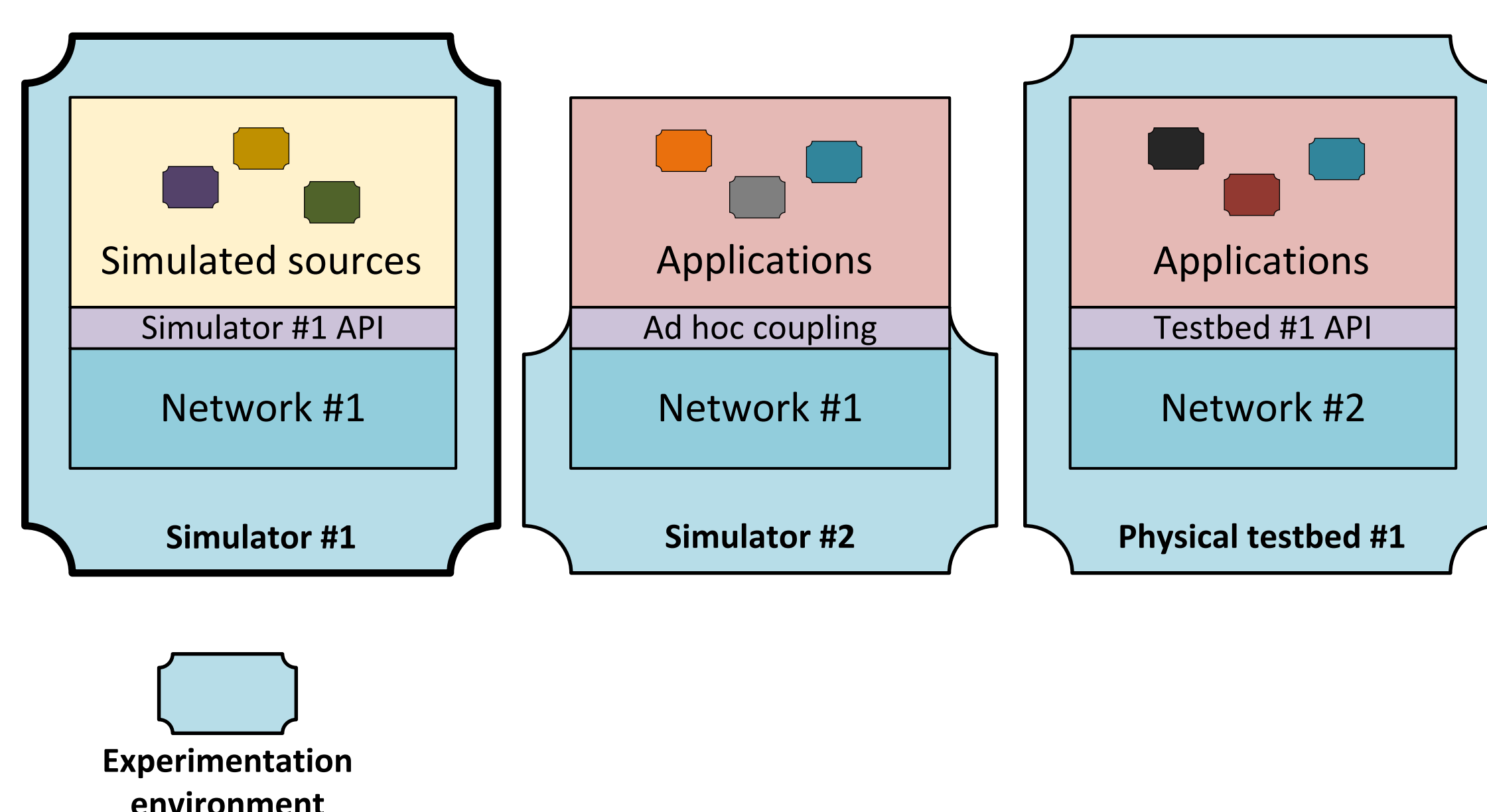


Figure 3. Environment-driven experimentation architectures

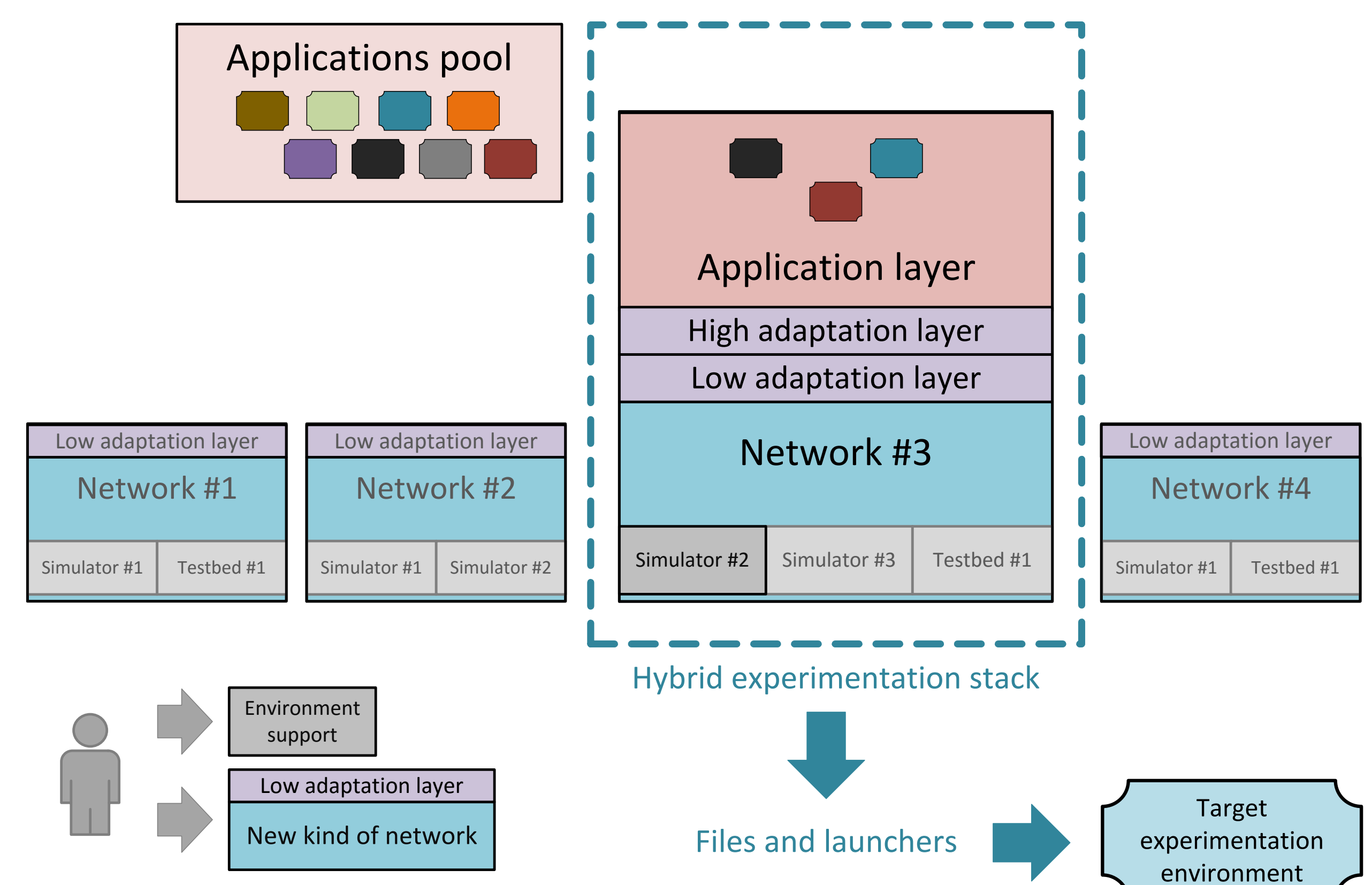


Figure 4. ERICA architecture

### Contact

Florian GREFF, PhD candidate  
 Thales Research & Technology  
 LORIA, University of Lorraine  
 florian.greff@loria.fr

[1] F. Greff, Y-Q. Song, A. Samama, L. Ciarletta, "Réseaux maillés dynamiques pour applications temps réel à criticités multiples: problématique et analyse", École Temps Réel, 2015

[2] F. Greff, E. Dujardin, A. Samama, Y-Q. Song, L. Ciarletta, "A Symbiotic Approach to Designing Cross-Layer QoS in Embedded Real-Time Systems", ERTS<sup>2</sup>, 2016

[3] F. Greff, Y-Q. Song, L. Ciarletta, A. Samama, "A Dynamic Protocol to the Design of a Software-Defined Real-time Mesh Network", IEEE WFCs, 2017